Risk and Portfolio Management with Econometrics, Courant Institute, Spring 2009
http://www.math.nyu.edu/faculty/goodman/teaching/RPME09/index.html
Always check the class bboard on the blackboard site from home.nyu.edu (click on academics, then on the course name) before doing any work on the assignment.

Assignment 8, due April 22, but Assignment 9 also will be due April 22

Corrections: (none yet)

Download the file `assignment8.xls` associated with this homework assignment. It contains four time series, each of which represents the daily closing price: the Euro/Dollar exchange rate (number of Dollars per Euro), the three month treasury yield (annualized), IBM, and an ETF[1] that represents the NASDAQ-100 index. All time series have about 500 entries. Because of trading differences, only IBM and QQQQ are quoted on exactly the same days.

For each of the series, C_t, (imagine that time is measured in days, so the time series is $X_1, X_2, \ldots, X_n$) you can construct the differences $X_t = C_{t+1} - C_t$. For the stocks, you also could consider returns $R_t = (X_{t+1} - X_t)/X_t$, or the log return $Y_t = \log(C_{t+1}/C_t)$. Read the data into Matlab and compute the differences and the returns. In what follows, a generic time series that might be invariant (see Meucci, Chapter 3) is called X_t. In each case, I want you to create a Matlab M-file that does the requested computation so that it is easy to experiment with many variations on them.

1. Create an M-file that performs one dimensional density estimation given n samples $X_1, \ldots, X_n$. If $f^e(x) = \frac{1}{n}\sum_{t=1}^{n} \delta(x - X_t)$, the estimated density is $f^e_\epsilon = \phi_\epsilon * f^e$ (the $*$ represents convolution), where $\phi_\epsilon(t) = \frac{1}{\epsilon}\phi(t/\epsilon)$, where ϕ is a mollifier. For these exercises, you can try ϕ to be a standard normal density, or a uniform density on the interval $[-1, 1]$. Apply your density estimator to each of the time series (to the X_t and C_t or Y_t according to your interests) and plot the results at a large number of uniformly spaced points, with spacing much smaller than ϵ. Plot the results on the interval $[\overline{X} - 5\sigma_X, \overline{X} + 5\sigma_X]$, where $\overline{X}$ and σ_X are the sample mean and standard deviation respectively. Choose values of $]\epsilon$ as small as possible without letting the estimated density become too noisy. Please explore these plots in some or all of the following ways:

   (a) How Gaussian are the estimated densities? You can determine this by plotting on the same plot, the Gaussian density with the same mean and variance.

   (b) How different are the densities from each other? You can determine this by plotting two or more of the curves on the same plot. As with the Gaussian, it will be important first to normalize to give them the same mean and variance.

[1] ETF stands for Exchange Traded Fund.

(c) How noisy are the plots? You can determine this by using the bootstrap method. Create a number (maybe five) of artificial resampled data-sets and estimate the density for each. If ϵ is large, the five estimated densities should be very similar. You know ϵ is too small when the estimated densities are significantly different.

2. Create an M-file that takes a time series, computes its mean and variance, and returns the number of entries, X_t, with $\left|X_t - \overline{X}\right| > k\sigma_x$, for a specified value of k. Choose values $k_{.1}$, $k_{.05}$, and $k_{.01}$, where $\Pr(|Z| > k_p) = p$ for a standard normal. Apply this to the various time series that may be invariants and comment on the relative frequency of large values compared to what they would be if the variables were Gaussian. Comment on the differences between the different time series.

3. Create an M-file that maps a given one variable data-set to the interval $[0, 1]$ using the empirical distribution function. That is, if N_t is the number of samples $X_s < X_t$, then $U_t = N_t/n$. Use this to create an M-file that takes a two dimensional data-set of points in the plane and maps it to another data-set of point in the unit square. This is described in Meucci's book. As a first application, we will look for serial correlation in QQQQ returns, that is $(X_t, Y_t) = (R_t, R_{t+1})$, where R_t is the daily QQQQ return. It is not clear how to test for independence between the U and V distributions. Here are some possibilities:

   (a) Create a 2D histogram. That is, divide the u and v axes into k equal intervals (or *bins*) so that the unit square is divided into k^2 equal sub-squares. Now count the number of (U_t, V_t) points in each of these squares and plot the 2D histogram (Matlab can make plots like that). The problem is that if k is large, then 500 data-points is too few to make accurate density estimates. For example, $k = 20$ leads to 400 boxes, so there will be an average of 1.2 points per box. Most boxes will have zero or one data-point. Do this with $k = 4$ instead and see whether the resulting 16 histogram counts are about the same (as they should be for independence). Can you construct a systematic statistical test to test the hypothesis that the 16 in counts are the same?

   (b) Use a two dimensional version of the Kolmogorov-Smirnov test. Suppose there are n points in the unit square. For any point (u, v) in the unit square, define $N(u, v)$ to be the number of data points in the rectangle below and to the left of (u, v), that is, the number of points (U_t, V_t) with $U_t < u$ and $V_t < v$. If they were uniformly distributed in the unit square, then we would have $N(u, v) \approx nuv$ (the total number of points times the area). Define the random variable M as $M = \max |N(u, v) - nuv|$, where the max is over all (u, v) in the unit square. Create an M-file to calculate M from a data-set.

   We want to test the hypothesis that the (U_t, V_t) are uniformly distributed in the square using the statistic M. If M is too large, we

will reject the hypothesis and conclude that the points are not uniformly distributed. For that, we need to know the distribution of M when the points are uniformly distributed. Do this by Monte Carlo. Generate L artificial set of n points that are uniform in the unit square. For artificial set j, use the M-files you have to compute M_j. Then find the 5%, and the 1% levels, called $m_{.01}$ and $m_{.01}$ respectively. You can reject the hypothesis that the actual datapoints are uniformly distributed at the 99% confidence level if the actual $M > m_{.01}$, where M is from the actual data. Use this test to test for serial correlation in the various time series.